Amendments to the Claims:

1. (Previously Presented) A search method comprising the steps of:
    (a)   partitioning a search region into n segments, wherein n is greater than 0;
    (b)   searching each segment with a first predetermined algorithm beginning at any location within at least one of the n segments;
    (c)   for each segment so searched, generating information indicating whether or not any indicator bit set to a predetermined state has been detected and the location of the indicator bit; and
    (d)   using the information provided in step c) to select a winning location.

2. (Original) The search method of claim 1 further including the step of performing a predetermined action on an entity associated with the winning location.

3. (Original) The search method of claim 1 or claim 2 further including the step of determining with a second algorithm a location in the segment from which searching starts.

4. (Original) The search method of claim 3 further including the step of providing a pointer to identify the location whereat searching begins; and

    stepping the pointer sequentially to access a plurality of locations within the region; and

    testing indicator bit at each location to see if it is set in the predetermined state; and

    generating a control signal for the first location encountered with the indicator bit set to the predetermined state.

5.   (Original) The method of claim 1 wherein the search region includes a plurality of contiguous locations to which information can be written or deleted and an indicator whose setting indicates information or no information at a selected location.

6.   (Original) The method of claim 5 wherein the information includes an identification number for at least one flow queue.

7.   (Previously Presented) A search method comprising the steps of:
     (a)   partitioning a search region into n segments, where n is greater than 0;
     (b)   searching the n segments simultaneously with a first predetermined algorithm;
     (c)   for each segment so searched, generating information indicating whether or not any indicator bit set to a predetermined state has been detected and the location of the indicator bit; and
     (d)   using the information provided in step c) to select a winning location.

8.   (Previously Presented) A search method comprising the steps of:
     (a)   partitioning a search region into n segments, wherein n is greater than 0;
     (b)   searching the n segments simultaneously with a first predetermined algorithm;
     (c)   for each segment so searched, generating information indicating whether or not any indicator bit set to a predetermined state has been detected and the location of the indicator bit; and
     (d)   using the information provided in step c) to select a winning location wherein for each of the n segments first it is assumed no starting point (CP) is in a segment being searched wherein searching begins at a first location of the segment and ends at a last location of said segment; and

second it is assumed a current pointer (CP) is in the segment being searched wherein searching begins at the CP location in said segment being searched ending at the last location of the segment and searching begins at the first location in the segment ending at the location preceding the CP.

9. (Previously Presented) A method to determine the next packet to forward from one of a plurality of flow queues comprising:

    (a)   providing in a memory a search region including a plurality of contiguous locations to which information can be written/deleted and an indicator whose state indicates the present or absent of information at a selected location;

    (b)   partitioning said search region into n segments, wherein n is greater than 0;

    (c)   determining a first location from which searching begins for each segment wherein said first location is not a fixed one but can be any location within a segment that is being searched;

    (d)   searching each segment in accordance with a predetermined algorithm;

    (e)   generating from each segment information indicating whether or not any indicator set to a predetermined state has been detected in said segment and location of detection;

    (f)   examining the information in (e) with a second predetermined algorithm to select a winner indicator and location.

10. (Previously Presented) g) The method in claim 9 further including using the information in step f) to move a packet from a queue associated with the location in step f).

11.   (Previously Presented) An apparatus including:

n traffic flow Queues, wherein n is greater than 0;

a processing complex including at least one processor that enqueues packets on selected ones of the traffic flow queues;

a memory with a search zone partitioned into a plurality of sectors wherein each sector includes a plurality of search locations with each search location including at least one indicator;

p segment search engines, p is greater than 1, and each of said p segment search engine includes m inputs wherein each one of the m inputs operatively coupled to an indicator within a group of indicators; and

a top search engine responsive to signals provided by the p segment search engines to generate a control signal identifying a location within said search zone wherein each of the p segment search engine is being assigned to search a segment beginning at any location within said segment.

12.   (Previously Presented) The apparatus of claim 11 further including

a first scheduler function that monitors the traffic flow queues and periodically attaches to a location in said search zone a characteristic associated with a traffic flow queue if a packet is placed in said traffic flow queue; and

a second scheduler function responsive to the control signal to transmit a packet from a selected Flow Queue.

13.   (Original) The apparatus of claim 12 further including a plurality of target port Queues wherein one of said target port queues received the transmitted packet.

14.   (Previously Presented) The apparatus of Claim 12 wherein the characteristic includes the flow queue identification number.

15.   (Previously Presented) A device comprising:

p segment search engines, p greater than 1 and each segment search engine having m inputs, m greater than 1, representing portions of a search zone;

a current pointer CP indicating a location whereat searching begins wherein said location being any one of the locations within said search zone;

at least one storage location that stores information outputted from each of the p segment search engines; and

a top search engine responsive to stored information to select and identify one of the locations in said search zone.

16.   (Original) The method of claim 1 wherein each segment includes m entries, wherein m is an even power of 2.

17.   (Original) The method of claim 1 or 16 wherein step d) further includes the step of correlating outputs from each segment search with a top search algorithm to select the winning location.

18.   (Previously Presented) A method for controlling the flow of information packets within a communications device including the steps of:

(a)   partitioning a calendar into n segments, wherein n is greater than 0;

(b)   searching each segment, beginning at any location within said each segment, with a segment search algorithm to identify at least one location with an indicator set to a first state;

(c)   examining with a top search algorithm locations detected in step (b); and

(d)   selecting one of the locations as a winning location.

19.　(Previously Presented) The method of Claim 18 further including the steps of determining a final winning location by concatenating an identification number for a winner segment containing the winning location to a value for the winning location within said winner segment; and

　　forwarding a packet from a flow queue having same identification number matching an identification number stored at the final winning location.

20.　(Previously Presented) A program product including:

　　a media on which a computer program is recorded, said computer program having

(a)　a set of instructions that partition a calendar into n segments, n is greater than 0;

(b)　a set of instructions that search each segment, beginning at any location within said each segment, with a segment search algorithm to identify at least one location with an indicator set to a first state;

(c)　a set of instructions that examines with a top search algorithm locations identified in step (b); and

(d)　a set of instructions that selects one of the locations as a winning location.

21.　(Original) The program product of Claim 20 further including

(e)　a set of instructions that determines a final winning location by concatenating an identification number for a winner segment containing the winning location to a value for the winning location within said winner segment; and

(f)　a set of instructions to generate and issue a signal that causes a device to forward a packet from a flow queue having an identification number matching an identification number stored at the final winning location.

22. (Previously Presented) A method to determine the next packet to forward from one of the plurality of flow queues comprising:

    (a)   providing in a memory a search region including a plurality of contiguous locations to which information can be written/deleted and an indicator whose state indicates the present or absent of information at a selected location;

    (b)   partitioning said search region into n segments, wherein n is greater than 0;

    (c)   determining a first location from which searching begins for each segment;

    (d)   searching each segment in accordance with a first predetermined algorithm wherein said searching assumes no current pointer (CP) is in a segment being searched wherein searching begins at a first location of the segment and ends at a last location of said segment; and said searching assumes a current pointer (CP) is in the segment being searched wherein searching begins at the CP location in said segment being searched ending at the last location of the segment and searching begins at the first location in the segment ending at a location preceding the CP;

    (e)   generating from each segment information indicating whether or not any indicator set to a predetermined state has been detected in said segment and location of detection;

    (f)   examining the information in (e) with a second predetermined algorithm to select a winner indicator and location.

23. (Previously Presented) The method of Claim 9 wherein for (d) first it is assumed no current pointer (CP) is in a segment being searched wherein searching begins at a first location of the segment and ends at a last location of said segment; and

second it is assumed a current pointer (CP) is in the segment being searched wherein searching begins at the CP location in said segment being searched ending at the last location of the segment and searching begins at the first location in the segment ending at the location preceding the CP.

24. (Cancel)

25. (Currently Amended) ~~The method of Claim 24~~ A method to select a queue from which a packet is to be transmitted within a communications device said method including:

  providing at least one calendar with n locations, n greater than 1, wherein each location includes id for a queue and an indicia to be set in one of two states;

  partitioning the n locations into a plurality of segments;

  searching each segment with a search engine to detect a first location having an associated indicia set to a predefined state;

  for each segment search, setting assumptions for a starting point CP whereat said searching begins wherein said assumptions cause multiple searches to be executed on at least one of said plurality of segment[[;]] ~~wherein~~ and the assumptions include no current pointer CP[[,]] is within a segment being searched wherein searching begins at a first location of the segment and ends at the last location of said segment[[;]] and a current pointer CP is in the segment being searched wherein searching begins at current pointer location in said segment being searched ending at the last location of said segment and searching begins at the first location of the segment ending at a location preceding the current pointer location[[.]];

  using outputs from the search engine to select one of the locations; and

  transferring a packet from the queue whose id is recorded at the one of the location.